RE '97 Third IEEE Int'l Symposium on Requirements Engineering

Annapolis, Maryland, USA

January 6-10, 1997

Preliminary Program

The Third IEEE International Symposium on Requirements Engineering (RE '97) provides a forum for researchers and practitioners to discuss requirements engineering, the branch of software engineering concerned with methods, techniques, and tools for eliciting, specifying, and analyzing software requirements. Featured this year is an industrial program consisting of industry experience reports, a panel on the requirements problem in industry, 5 tutorials, and a tools exhibit. The symposium also features 21 research papers, a minitutorial, 2 workshops, a panel on change, and a doctoral consortium. The symposium site is historic downtown Annapolis, 30 miles east of Washington, D.C.

SCHEDULE

Monday (Jan. 6): Doctoral Consortium; Monday, Tuesday (Jan. 6-7): Tutorials; Wednesday, Thursday, Friday (Jan. 8-10): Technical Program

Keynote Speakers

Anthony Hall *(Praxis)* **"What's the Use of Requirements Engineering?"**
Many approaches to requirements engineering exist but often conflict. Conflicts can best be resolved by asking: "What is the use of doing that?". How addressing this question helps in choosing requirements methods and in dealing with difficulties that arise in applying the methods is discussed. Dr. Hall, a principal consultant with the software engineering company Praxis, pioneered the use of formal specification in industrial projects and led the design of the CDIS air traffic information system, one of the largest industrial applications of formal methods. He has worked on requirements for many systems and guided the development of major systems from requirements.

Colin Potts *(Georgia Inst. Tech.)* **"Requirements Models in Context"**
Traditional requirements engineering stresses generalization and abstraction. But, by abstracting away from the context, the designer may model only those things that are easy to model and ignore the subtleties, special cases, and concrete features of the context. In contrast, approaches that stress context at the expense of abstraction may lead to floundering or to short-term customer satisfaction at the expense of long-term system fragility. Needed is a synthesis of the two approaches. Professor Potts, a member of Georgia Tech's Software Research Ctr. and its Graphics, Visualization and Usability Ctr., has held positions in both industrial R&D and software development.

John Rushby *(SRI International)* **"Calculating with Requirements"**
Formal techniques, such as strong type checking and completeness and consistency checking using decision procedures and model checking, reduce certain questions about requirements to automated (and therefore fast, cheap, and repeatable) calculations. Examples from space shuttle and other applications illustrate the techniques. Dr. Rushby, Program Director of SRI's Computer Science Lab., develops formal verification systems (the latest is PVS) and applies them to problems in computer security, hardware design, and safety-critical and fault-tolerant systems. PVS is currently being used in industrial projects applying formal methods to aerospace problems.

David Harel *(Weizmann Institute of Science)* **"Will I Be Pretty, Will I Be Rich? Theory vs. Practice in Systems Engineering"**
What is the role of theoretical vs. applied research in the specification and design of reactive, highly concurrent systems? This talk classifies the research performed by theoreticians into three kinds of theory—theory for the sake of theory, theory of foundations and principles, and theory arising from applications. The different kinds of theory are illustrated with examples from several areas of computer science. Professor Harel is the William Sussman Professor of Mathematics at the Weizmann Institute. A cofounder and chief scientist of i-Logix, Inc., he is also the inventor of the statecharts language and was part of the team that designed the Statemate system. His most recent book is "Algorithmics: The Spirit of Computing" (MacMillan 1988).

Minitutorial

Model Checking and Requirements - Daniel Jackson *(Carnegie Mellon University).* With its dramatic success in automatically detecting design errors (mainly in hardware and protocols), model checking has recently rescued the reputation of formal methods. This tutorial describes what model checking is, what tools have been developed, and how the tools might be used to analyze requirements. It also introduces *model enumeration*, a new technique that, unlike model checking, allows structures, rather than event sequences, to be analyzed automatically.

Panels

Impact of Environmental Evolution on Requirements Changes - Chair: Nazim Madhavji *(McGill University).* When a system is being developed, the system's environment usually keeps evolving. This environmental evolution may adversely affect the system implementation, causing functional deficiencies, performance problems, etc. To avoid such problems, the effects of environmental changes on system requirements must be identified. This panel will discuss the impact of environmental change on requirements and how this problem can be understood and solved.

Industrial Priorities for Requirements Engineering Research - Chair: Steve Miller *(Rockwell-Collins).* Solutions to problems studied by the RE research community often do not meet the real needs of industry. Industry representatives will provide insight into the most critical problems that research should be attacking.

Tutorials

Making Requirements Measurable — Bashar Nuseibeh *(Imperial College)* and **Suzanne Robertson** *(Atlantic Systems)* Participants in this full-day "interactive" tutorial examine measurability by building a requirements specification for a familiar system. A requirements template is used as a guide. How measurable requirements can be used to build a requirements quality filter is described.

Requirements Specification and Analysis With SCR — Stuart Faulk *(University of Oregon)* and **Connie Heitmeyer** *(Naval Research Laboratory).* This half-day tutorial describes the practical, industrial-strength Software Cost Reduction (SCR) method for developing requirements. The formal model that underlies SCR and software tools supporting consistency checking, simulation, and verification are described. The application of SCR to two practical systems is discussed.

Software Requirements Specification and System Safety — Mats Heimdahl *(University of Minnesota)* and **Jon Reese** *(University of Washington).* After introducing system safety, this half-day tutorial discusses how software control affects safety analysis and outlines the root causes of safety problems. The formal language RSML (Requirements State Machine Language) is introduced. RSML has been used to capture the requirements of several safety-critical systems, most notably TCAS II.

Requirements Traceability — Anthony Finkelstein *(City University, London)* and **Richard Stevens** *(QSS).* This half-day tutorial focuses on requirements traceability, the ability to describe and follow information about the life of a requirement. The focus will be on traceability in a systems engineering setting. The tutorial will provide a detailed look at requirements traceability and practical techniques for supporting it.

Object-Oriented Requirements Specification — Roel Wieringa *(Free University, The Netherlands).* This half-day tutorial presents the latest object-oriented requirements methods and compares them to recent developments in structured analysis. Four methods are covered: the Unified Modeling Language of Rumbaugh, Booch and Jacobson; Fusion (1996) extended with use cases; OOA (Shlaer-Mellor); and Yourdon Systems Method (1993). The potential for combining different methods is discussed.

Workshops

Scenario-Based RE Methods. While scenarios have become an important component of requirements engineering, little guidance exists on how scenarios may be used in validation, requirements elicitation, etc. This workshop explores the different concepts of scenarios and whether a common view of scenarios exists. Means of technology transfer and research challenges will be discussed. (Organized by **Alistair Sutcliffe,** *City University, London)*

Software on Demand: Issues for RE. Software on demand is software that can be delivered over the Internet on an as-needed basis. The user can download full applications or small plug-ins to complete the current task at hand. This workshop will explore topics such as how to specify the requirements of software on demand and how software on the net can be organized. A prototype software on demand system will be used as a strawman. (Organized by **Steve Fickas,** *University of Oregon)*

Tools Exhibit

Chairs: Charles Payne, Dwight Colby *(Secure Computing Corp.).* Presentations and demos of state-of-the-art commercial tools along with cutting edge academic efforts are scheduled. Confirmed exhibitors include Vitech Corp. (CORE), Marconi Systems Technology, Inc. (RTM), QSS (DOORS), TD Technology (SLATE), Universite' Catholique de Louvain (GRAIL/KAOS), and Naval Research Lab (SCR Toolset).

Doctoral Consortium

Chair: Myla Archer *(Naval Research Lab).* One of the most popular events of RE '95, the Doctoral Consortium gives students whose doctoral research is not yet complete an opportunity to present their work to colleagues in RE.

Organizing Committee

General Chair: Connie Heitmeyer *(Naval Research Lab)* heitmeyer@itd.nrl.navy.mil
Program Chair: John Mylopoulos *(University of Toronto)* jm@cs.toronto.edu
Industrial Chair: Stuart Faulk *(University of Oregon)* faulk@cs.uoregon.edu

More information and registration materials:
http://www.itd.nrl.navy.mil/conf/ISRE97

IEEE COMPUTER SOCIETY 50 YEARS OF SERVICE •1946-1996